OIPE
APR 26 2004
PATENT & TRADEMARK OFFICE
JC95

Docket No. AUS990339US3

PATENT

#14

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

RECEIVED
APR 28 2004
Technology Center 2100

In re application of: **Bahrs et al.**

Serial No.: **09/429,522**

Filed: **October 28, 1999**

For: **Method and Apparatus in a Data Processing System for Systematically Separating Application Graphical User Interface Component Placement from Component Sequencing and Compound Creation**

Group Art Unit: **2178**

Examiner: **Paula, Cesar B.**

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

ATTENTION: Board of Patent Appeals and Interferences

Certificate of Mailing Under 37 C.F.R. § 1.8(a)
I hereby certify this correspondence is being deposited with the United States Postal Service as First Class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on 04/23/04.

By: Amelia C. Turner
Amelia C. Turner

05/06/2004 SCHAPMAN 00000002 090447 09429522
01 FC:1402 330.00 DA

APPELLANT'S BRIEF (37 C.F.R. 1.192)

This brief is in furtherance of the Notice of Appeal, filed in this case on January 23, 2004.

The fees required under § 1.17(c), and any required petition for extension of time for filing this brief and fees therefore, are dealt with in the accompanying TRANSMITTAL OF APPEAL BRIEF.

This brief is transmitted in triplicate. (37 C.F.R. 1.192(a))

REAL PARTIES IN INTEREST

As reflected in the Assignment recorded on October 28, 1999, at Reel 010348, Frame 0928, the present application is assigned to International Business Machines Corporation, the real party in interest.

RELATED APPEALS AND INTERFERENCES

With respect to other appeals or interferences that will directly affect, or be directly affected by, or have a bearing on the Board's decision in the pending appeal, there are no such appeals or interferences.

STATUS OF CLAIMS

Claims 64-75 and 77-82 stand finally rejected by the Examiner as noted in the final rejection mailed October 23, 2003.

STATUS OF AMENDMENTS

Applicants' Supplemental Response to Office Action, transmitted on January 23, 2004, has been entered.

SUMMARY OF INVENTION

According to claim 64, Applicants claim a method for displaying a component or container. A component is displayed within a display using a first object. Responsive to receipt within the first object of user input, the first object sends the user input to a third object. A second object controls location of the component within the display. The second object controls the location of the component in response to receiving an event from the third object. Response to the third object

receiving the user input, the third object is used to determine whether a change in the location is required. Responsive to a determination by the third object that the change in location is required, the third object is used to selectively display the component by generating the event by the third object. The event indicates that the location is to be changed.

According to claim 66, Applicants claim a method for displaying a graphical user interface. A container is displayed for a graphical user interface using a view controller object. Responsive to a receipt within the view controller of user input, the view controller sends the user input to an application mediator object. A placement object controls the location of the container. The placement object controls the location of the container in the graphical user interface in response to receiving an event from the application mediator object. Responsive to the application mediator object receiving the user input, the application mediator object uses the user input to determine whether a change in the location is required. Responsive to a determination by the application mediator object that the change in location is required, the application mediator object is used to generate the event. The event indicates that the location is to be changed. The event is sent to the placement object.

ISSUES

Are the Examiner's rejections of claims 64, 66-71, 73-74, 76-77, and 79-82 under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent 6,330,659 issued to *Poff* in view of U.S. Patent 6,054,985 issued to *Morgan*; and of claims 65, 72, 75, and 78 under 35 U.S.C. § 103(a) as being unpatentable over *Poff* in view of *Morgan* and further in view of "Mastering JavaBeans", Chapter 3, Vanhelsuwe well founded?

GROUPING OF CLAIMS

For the purposes of this appeal, claims 64-75 and 77-82 stand or fall together as one group.

ARGUMENT

The Examiner rejected claims 64, 66-71, 73-74, 76-77, and 79-82 under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent 6,330,659 issued to *Poff* in view of U.S. Patent 6,054,985 issued to *Morgan*. This position is not well founded.

The combination of references cited by the Examiner does not render Applicants' claims unpatentable. The combination fails to disclose receipt within the first object of selected user input, sending user input from a first object to a third object, the manner in which the elements of Applicants' claims interact with each other, determining by a third object whether a change in location is required, or a third object generating an event to send to a second object that indicates that the location is required to be changed.

Applicants claim displaying a component within a display using a first object. The Examiner states that the first object is the windowing system task processor of *Poff* as described in column 4, line 42, through column 5, line 27; column 15, lines 56-67; Figure 7; and Figure 11.

Applicants' claim next describes "responsive to a receipt within the first object of selected user input, sending, by the first object, the user input to a third object". The Examiner states that *Poff* describes sending user input requests from the task processor to the accelerator, where the accelerator is the third object as described in column 12, lines 25-31, and column 15, lines 56-67. The Examiner disregards the first part of this element of the claim, however, which states "responsive to a receipt within the first object of selected user input". Nothing in *Poff* describes the task processor receiving user input. In fact, *Poff* describes the Java AWT accelerator being used to manage user input such as mouse clicks and keystrokes. See column 3, lines 2-3 and column 12, lines 25-31. *Poff* does not describe the first object receiving user inputs. If the accelerator is considered to be the third object of Applicants' claims, it is the third object according to *Poff* that receives user inputs.

The Examiner also relies on column 12, lines 25-31. This part of *Poff* describes the accelerator being used to draw windows and buttons and can also be used to manage user input such as mouse clicks. This is further clarified in the next sentence in the reference in column 12, lines 29-31, which states that the accelerator is provided an interface to the display, keyboard, and mouse. Therefore, it is quite clear that the accelerator is what receives user inputs. The

Examiner states that the accelerator is the third object while the task processor is the first object. Thus, according to the Examiner's reading of the reference, it is the third object, not the first object, that receives user inputs.

Applicants' claim then describes the first object sending the user input to the third object. Again, according to the Examiner, the task processor sends user inputs to the accelerator which teaches the first object sending user inputs to the third object. However, *Poff* does not describe the task processor sending user inputs to the accelerator. *Poff* describes the task processor directing object requests to the accelerator. These object requests are not described as being user inputs. "Such requests can include; create a container 110; create a borderlayout 112; create a panel 114; create a label 116; create a textfield 118; and create a button 120." These object requests are not user inputs. Therefore, *Poff* does not teach the task processor sending user inputs to the accelerator.

According to the Examiner, *Poff* teaches the first object by teaching the task processor and teaches the third object by teaching the accelerator. If the first object is considered to be the task processor and the third object is considered to be the accelerator, *Poff* does not teach "responsive to a receipt within the first object of selected user input" because *Poff* does not teach a first object, the task processor according to the Examiner, receiving user input. *Poff* does not teach "sending, by the first object, the user input to a third object" because *Poff* does not teach the task processor sending user inputs to the accelerator.

The Examiner combines *Morgan* with *Poff* to add features the Examiner states that *Poff* does not disclose. Applicants claim a combination of elements. Further, the claims describe how these elements interact with each other. How the elements of the claims are combined and interact with each other is itself an important feature of the claims. For example, Applicants claim the second object controlling the location of the component in response to receiving an event from the third object which determined that a change in location is required. Combining *Poff* with *Morgan* does not describe, teach, or suggest the combination claimed by Applicants of displaying a component using a first object that receives user input and sending that user input to a third object, which then determines whether a change in location is required; and if a change in location is required, the third object selectively displaying the component by generating an event that is received by the second object which then controls the location of the component. The

combination of *Poff* and *Morgan* does not describe, teach, or suggest this combination of features.

The Examiner states that *Poff* does not teach controlling a location using a second object where the second object controls the location in response to receiving an event from the third object. The Examiner cites *Morgan* to supply this missing feature. The Examiner states that *Morgan* teaches moving a window to a new location by dragging and dropping the window. However, the Examiner does not point to any section of either reference that describes controlling the location in response to receiving an event from a third object.

The Examiner states that *Poff* does not teach using by the third object the user input to determine whether a change in location is required. The Examiner again points to *Morgan* to supply this missing feature. The combination of references, however, does not describe, teach, or suggest determining whether a change in location is required. The combination also does not describe, teach, or suggest a third object using user input received from a first object to determine whether a change in location is required.

The Examiner states that *Poff* does not teach responsive to a determination by the third object that the change in location is required, selectively displaying the component using the third object by generating the event by the third object where the event indicates that the location is to be changed. Again, the Examiner uses *Morgan* to supply this feature stating that *Morgan* teaches moving windows. However, the combination of references does not describe, teach,, or suggest a third object sending an event to a second object where the event indicates that the location is to be changed.

The combination of references does not describe, teach, or suggest Applicants' claims. *Poff* does not describe, teach, or suggest a receipt within the first object of selected user input. *Poff* does not describe, teach, or suggest sending user input from a first object to a third object. The combination of *Poff* and *Morgan* does not describe, teach, or suggest the manner in which the elements of Applicants' claims interact with each other. The combination does not describe, teach, or suggest determining by a third object whether a change in location is required. The combination does not describe, teach, or suggest a third object generating an event to send to a second object that indicates that the location is required to be changed.

The Examiner rejected claims 65, 72, 75, and 78 under 35 U.S.C. § 103(a) as being unpatentable over *Poff* in view of *Morgan* and further in view of "Mastering JavaBeans", Chapter 3. This position is not well founded.

These claims describe the first object being a view controller, the second object being a placement listener, and the third object being an application mediator.

The Examiner states that *Poff* fails to disclose a placement listener stating that "Mastering JavaBeans" discloses listeners. The combination of these references, however, fails to disclose receipt within the first object of selected user input, sending user input from a first object to a third object, the manner the elements of Applicants' claims interact with each other, determining by a third object whether a change in location is required, or a third object generating an event to send to a second object that indicates that the location is required to be changed, where the first object is a view controller, the second object is a placement listener, and the third object is an application mediator.

The combination of references cited by the Examiner does not render Applicants' claims unpatentable. The combination fails to disclose receipt within the first object of selected user input, sending user input from a first object to a third object, the manner in which the elements of Applicants' claims interact with each other, determining by a third object whether a change in location is required, or a third object generating an event to send to a second object that indicates that the location is required to be changed.

Lisa L.B. Yociss
Reg. No. 36,975
Yee & Associates, P.C.
PO Box 802333
Dallas, TX 75380
(972) 367-2001

APPENDIX OF CLAIMS

The text of the claims involved in the appeal reads:

64. A method in a data processing system for displaying a component or container, the method comprising the data processing system implemented steps of:

displaying the component within a display using a first object;

responsive to a receipt within the first object of selected user input, sending, by the first object, the user input to a third object;

controlling a location of the component within the display using a second object, wherein the second object controls the location of the component in response to receiving an event from the third object;

responsive to the third object receiving the selected user input, using, by the third object, the user input to determine whether a change in the location is required; and

responsive to a determination by the third object that the change in the location is required, selectively displaying the component using the third object by generating the event by the third object, the event indicating that the location is to be changed, wherein the third object generates the event.

65. The method of claim 64, wherein the first object is a view controller, the second object is a placement listener, and the third object is an application mediator.

66. A method in a data processing system for displaying a graphical user interface, the method comprising:

displaying a container for a graphical user interface using a view controller object;

responsive to a receipt within the view controller of selected user input, sending, by the view controller, the user input to an application mediator object;

controlling a location of the container using a placement object, wherein the placement object controls the location of the container in the graphical user interface in response to receiving an event from the application mediator object;

responsive to the application mediator object receiving the selected user input, using, by the application mediator object, the selected user input to determine whether a change in the location is required; and

responsive to a determination by the application mediator object that the change in the location is required, generating the event using the application mediator object, the event indicating that the location is to be changed, wherein the event is sent to the placement object.

67. The method of claim 66, wherein the container is a panel.

68. The method of claim 66, wherein the container is a button.

69. A display mechanism for use in a data processing system to display a container in a display in the data processing system, the display mechanism comprising:

a first object used to display a graphical user interface in the display and to receive user input;

response to a receipt of said user input by the first object, said first object for sending the user input to a third object;

a second object used to control a location of the graphical user interface in the display in response to receiving an event from the third object;

the third object for using the user input to determine whether a change in the location is required in response to the third object receiving the selected user input and

the third object for generating the event responsive to a determination by the third object that the change in the location is required, the event indicating that the location is to be changed.

70. The display mechanism of claim 69, wherein the first object is a display object and the second object is a positioning object.

71. The display mechanism of claim 70, wherein the display object is an instance of a view controller.

72. The display mechanism of claim 70, wherein the positioning object is an instance of a placement listener.

73. The display mechanism of claim 69, wherein the display mechanism is implemented in Java.

74. The display mechanism of claim 69, wherein the second object is useable with a plurality of first objects.

75. The display mechanism of claim 69, wherein the first object is a view controller, the second object is a placement listener, and the third object is an application mediator.

77. A data processing system for displaying a component or container, the system comprising:

displaying means for displaying the component within a display using a first object;

sending means for sending user input from the first object to a third object responsive to a receipt within the first object of the user input;

controlling means for controlling a location of the component within the display using a second object, wherein the second object controls the location of the component in response to receiving an event from the third object;

determining means responsive to the third object receiving the user input, for using the user input to determine, by the third object, whether a change in the location is required; and

displaying means responsive to a determination by the third object that the change in the location is required, for selectively displaying the component using the third object by generating the event by the third object, the event indicating that the location is to be changed, wherein the third object generates the event.

78. The system of claim 77, wherein the first object is a view controller, the second object is a placement listener, and the third object is an application mediator.

79. A data processing system for displaying a graphical user interface, the system comprising:

displaying means for displaying a container for a graphical user interface using a view controller object;

sending means responsive to a receipt within the view controller object of selected user input, for sending, by the view controller object, the user input to an application mediator object;

controlling means for controlling a location of a container using a placement object, wherein the placement object controls the location of the container in the graphical user interface in response to receiving an event from the application mediator object;

determining means responsive to the application mediator object receiving the selected user input, for using the user input to determine, by the application mediator object, whether a change in the location is required; and

generating means responsive to a determination by the application mediator object that the change in the location is required, for generating the event using the application mediator object, the event indicating that the location is to be changed, wherein the event is sent to the placement object.

80. The system of claim 79, wherein the container is a panel.

81. The system of claim 79 wherein the container is a button.